

Amendments to the Specification:

Please replace the improperly formatted heading “**Related Application Information**” on page 1, prior to paragraph [01] with the following new section heading:

CROSS REFERENCE TO RELATED APPLICATIONS

Please replace the improperly formatted section heading “**Technical Field**” on page 1, prior to paragraph [02] with the following new section heading:

TECHNICAL FIELD

Please replace the improperly formatted section heading “**Background of the Invention**” on page 1, prior to paragraph [03] with the following new section heading:

BACKGROUND OF THE INVENTION

Please replace the improperly formatted section heading “**Summary of the Invention**” on page 3, prior to paragraph [08] with the following new section heading:

SUMMARY OF THE INVENTION

Please replace the improperly formatted section heading “**Brief Description of the Drawings**” on page 4, prior to paragraph [14] with the following new section heading:

BRIEF DESCRIPTION OF THE DRAWINGS

Please replace the improperly formatted section heading “**Detailed Description of the Invention**” on page 5, prior to paragraph [27] with the following new section heading:

DETAILED DESCRIPTION OF THE INVENTION

Please replace paragraph [34] with the following new paragraph:

Media objects may be supplied from a storage medium, such as a CD-ROM or a PC card, etc., or supplied directly from a capture device such as a camera or a scanner, or from the Internet or some other supplysource. Media objects are often stored in a database. Any database suitable for storing media data may be used. Cataloging refers to forming an index of a collection of media objects. Indexing refers to building a table or index that includes pointers to data. These pointers permit quick access to data without having to search and sort through information. Examples of index types that may be useful in the present invention include an index of metadata sorted by date, by photographer or other capturer, by longitude, by latitude, by zip code, by proximity to other objects, by subject matter or any other useful piece of information. Typical indexing methods include those used by databases, for example the known Indexed Sequential Access Method (ISAM). A catalog can be built as part of a file system or a database as appropriate for the desired method of access. As an example of a database, Microsoft Access by Microsoft Corp. Corporation of Redmond, Washington can be used. Microsoft Access which provides for a data store, indexing on multiple keys in order to provide quick access to sets of data and methods of performing a query based on multiple factors to generate a report. The categories for the database may be standard categories within the cataloging system and/or categories that are set by the user. For example, the database may include categories for cataloging the media objects such as particular holidays, vacations, locations, dates, and the like.

Please replace paragraph [40] with the following new paragraph:

Data collection includes gathering ancillary data before, during or after the capture of a media object. This may be done automatically or manually, and may be related to or result from another media object capture. The data may include information added by the user via an input device such as a keyboard, or it may be supplied from other external devices. For example, information from a GPS device indicating the location where the image was captured may be added, or audio may be added from memory or an external device. The user may add a textual description of the image. The ancillary data may be stored along with the media object. The media object database may be physically the same or different from the database for storing the information concerning

the media object. Underlying database technologies can include any general database, for example SQL Server, Access or Exchange provided by Microsoft ~~Corp.~~Corporation of Redmond, Washington and as known in the art.

Please replace paragraph [59] with the following new paragraph:

The process for automatically organizing media objects, such as still images, into various collections according their creation times will be described with reference to Fig. 9. In step 900, the images in some predetermined directory structure, such as the “My Pictures” directory under the Windows® operating system (OS), for example, are sorted in ascending order by creation date. In step S901, a variable denoted “lasttime” is set to effective negative infinity (e.g., a quantity less than the earliest date represented under the OS). An empty current collection is created in step S902. The first (earliest created) image is selected for processing in step S903. Of course, processing could be performed beginning with the latest (last created) image. The creation time of the current image is compared to the lasttime variable in step S904. If the difference between these two times is less than X hours, for some value of X, the two images are placed in the same collection. The value of X may be set by the user to any suitable time such as 1 hour, 24 hours, 15 minutes, etc. The shorter the time set for X, the more collections that are created, which will tend to be more thematically related, but will require the user to scan through more collections. A default value for X may be set and the user may adjust the default value as desired. If the answer in step S904 is Yes, then the image is stored in the current collection in step S905. If the answer in step S904 is No, then the current image is not stored in the current collection. Rather, a new empty current collection is created in step S910, and processing continues to step S905 where the current image becomes the first image stored in the new current collection.

Please replace paragraph beginning on page 20, line 8 (immediately before paragraph [63]) with the following new paragraph:

Referring to Fig. 10, a flowchart is provided which describes a color clustering algorithm for media objects (photographs in the present example). Color clustering is useful when the creation

time of the media objects are either unreliable or unavailable. In S1001, a color histogram for all images is created. The process of extracting color histograms is well known in the art and therefore will not be described in detail here. Preferably, the color histograms are taken over all pixels of images that are downsampled by a factor of 4 or 8, thus increasing the speed of the histogram computation. Optimally, the color histogram is taken in the CIE u' v' color space. By taking the color histogram in the CIE u' v' color space, pixel luminance is ignored. This is desirable because the pixel luminance does not predict the topic of the media object, here a photograph. Ideally, the histogram has 16×16 equally spaced bins in color space, spanning from 0.1612 to 0.2883 in u' and 0.4361 to 0.5361 in v' . Pixels falling outside this area in color space are mapped to the nearest bin.

Please replace paragraph [63] with the following new paragraph:

The histogram may be estimated in a Bayesian manner, using *maximum a posteriori* methods and a uniform prior, as is well known in the art. The histogram is initialized with so-called “phantom data.” Before all pixels of an image are scanned, every bin is initialized to 1/256. As is also known in the art, a “number of pixel” counter is required to compute a histogram; the number of pixel counter is initialized ~~the~~to 1. By initializing the number of pixel counter to 1, no bin is allowed to attain a value of 0 which is desirable in the computation of the Jensen-Shannon distance in equation 4 of S1004, described below. The histogram is best estimated in a smoothed manner as is further known in the art. The histogram may be smoothed via so-called bilinear interpolation wherein every scanned pixel increases up to four histogram bins, according to standard bi-linear interpolation rules.

Please replace paragraph [73] with the following new paragraph:

Referring to Fig. 11, a flowchart is shown depicting a method which combines temporal and color clustering of photographs. This combination uses the better clustering of the temporal algorithm but falls back to color clustering when the temporal data is incorrect. In step S1101, the pictures are sorted by creation date, ~~similar to steps S1101 and S1102 in Fig. 11~~. In step S1102, the temporal clustering algorithm is applied, as shown in Fig. 9 (except for S908 and

S909) resulting in a number of clusters. These clusters are then scanned in steps S1103 – S1107 to check for failure of the temporal clustering. The existence of overly large clusters is evidence of temporal clustering failure. Thus, step S1103 first sets the current cluster to the first cluster and step S1104 counts the number of images in the current cluster. If the number is larger than a threshold X, step S1105 will split the current cluster into sub-clusters according to the color algorithm described in Fig. 10, except for steps S1007 and S1008. Preferably, the threshold X is 23. If the number is larger than the threshold X, the desired number of sub-clusters is preferably 1/12 of the number of photographs in the current cluster, rounded to the nearest integer. Upon successful splitting of the current cluster into sub-clusters, or if the current cluster has less than or equal to X images, control flows to step S1106 which can terminate the loop if the current cluster is the last cluster. If it is not, step S1107-S1110 makes the current cluster the next cluster and control returns to step S1104. Otherwise, control flows to step S1108-S1107 which computes representative images for all of the clusters (including both temporal and color clusters). This step is analogous to step S1007 in Fig. 10. In step S1108, the representative images are shown in the user interface and in step S1109, the process is completed. It is desirable to sort the representative images by increasing creation time prior to display.